

A Concise Force Calculation for Hybrid Monte Carlo with Improved Actions

Nikhil Karthik*

*Department of Theoretical Physics,
Tata Institute of Fundamental Research,
Homi Bhabha Road, Mumbai 400005, India.*

We present a concise way to calculate force for Hybrid Monte Carlo with improved actions using the fact that changes in thin and smeared link matrices lie in their respective tangent vector spaces. Since hypercubic smearing schemes are very memory intensive, we also present a memory optimized implementation of them.

PACS numbers: 11.15.Ha, 12.38.Gc

I. INTRODUCTION

A standard method for dynamical simulation of QCD is by simulating an equivalent micro-canonical ensemble of a fictitious classical system [1]. Another method for global updates of gauge fields is through stochastic evolution using Langevin dynamics [2]. However, in both these methods, the finite step size used for the evolution through the simulation time introduces error. Hybrid Monte Carlo (HMC) [3] avoids this problem by combining molecular dynamics with Langevin-type refreshment of the momenta conjugate to the gauge links at the beginning of every trajectory and an acceptance step in the end. Such a micro-canonical evolution using classical dynamics requires the calculation of force, which in this case is the derivative of the Hamiltonian with respect to the gauge links.

Nowadays, gauge link smeared actions are commonly used in dynamical simulations to improve scaling behaviour, especially after the advent of stout [4] and HEX [5] schemes due to their differentiability. These actions are explicit functions of the smeared links. A method to calculate force for HMC with these improved actions was first discussed in [6], where a chain rule was used to find the variation of action with respect to the original thin-links. Using this chain rule, the force for HEX improved fermions was derived in [7], combining the methods utilised in [8] (for the “hypercubic” part of HEX) and [4] (for the “exp” part of HEX).

In this work, we show the simplicity of finding the force using the ordinary derivative with respect to a single real parameter, which is the magnitude of displacement in the tangent vector space of a gauge link. We take this method further to re-derive the force for the HEX improved HMC. For this, we note that a variation in a thin-link, which lies in its tangent vector space, causes variations in smeared links which also lie in their respective tangent vector spaces. With this observation, we again have to find directional derivatives with respect to a single real parameter in each of the tangent vector spaces of the smeared links. The unknowns are the directions in these tangent vector spaces, which are relatively easy to find. This method differs substantially from the one used in [7] by not requiring to find derivatives of action with respect to each matrix element of thin and smeared links. We find that this makes the calculation easy to check and that it can be easily extended to other nested improvement schemes, which would otherwise be difficult due to the constant book-keeping of colour indices required.

This work is based on the calculation of HEX force in [7] and we borrow their notation as much as possible. In Section II, we introduce our notation and describe the construction of HEX smeared quarks. Section III deals with finding the derivatives of functions defined on an $SU(N_c)$ manifold. Using the methods developed, we give a concise derivation of the HEX force in Section IV. In Section V, we give a memory optimized implementation of hypercubic smearing schemes.

II. DEFINITIONS

The lattices contain N_t points along the temporal direction and N_s points in the spatial directions. The volume of the lattice is $V_{lat} = N_t N_s^3$. A point on this lattice is labelled by the variables x, y and l . The directions are labelled by Greek indices and their values run from 1 to 4. In this notation, the gauge link connecting a point x to $x + \mu$ is written as $U_{x,\mu}$. The Einstein summation convention is not used and summations over indices are indicated explicitly.

*Electronic address: nikhil@theory.tifr.res.in

To be concrete, we deal with N_f flavours of staggered quarks. The standard staggered Dirac operator is constructed out of thin-links, U , and it is given by

$$D_{x,y} = m\delta_{x,y} + \frac{1}{2} \sum_{\nu=1}^4 \left(U_{x,\nu} \delta_{x,y-\nu} - U_{x-\nu,\nu}^\dagger \delta_{x,y+\nu} \right), \quad (1)$$

where the gauge links have been pre-multiplied with the staggered phases, $\eta_{x,\nu}$, and m is the bare quark mass in lattice units.

A single level of HEX has three nested sub-levels constructed such that the smeared links at the n -th sub-level, $V^{(n)}$, are built only out of the thin-links within a hypercube. The final HEX smeared links, $V_\mu^{(3)}$ obtained in the third sub-level, are given by [5]

$$\begin{aligned} \text{Sub-level 1: } V_{x,\mu;\nu\rho}^{(1)} &= \exp \left(\left[C_{x,\mu;\nu\rho}^{(1)} U_{x,\mu}^\dagger \right]_{\text{TA}} \right) U_{x,\mu}, & C_{x,\mu;\nu\rho}^{(1)} &= \frac{\epsilon_1}{2} \sum_{\pm\sigma \neq \mu\nu\rho} \left[U_{x,\sigma} U_{x+\sigma,\mu} U_{x+\mu,\sigma}^\dagger \right] \\ \text{Sub-level 2: } V_{x,\mu;\nu}^{(2)} &= \exp \left(\left[C_{x,\mu;\nu}^{(2)} U_{x,\mu}^\dagger \right]_{\text{TA}} \right) U_{x,\mu}, & C_{x,\mu;\nu}^{(2)} &= \frac{\epsilon_2}{4} \sum_{\pm\sigma \neq \mu\nu} \left[V_{x,\sigma;\mu\nu}^{(1)} V_{x+\sigma,\mu;\sigma\nu}^{(1)} V_{x+\mu,\sigma;\mu\nu}^{(1)\dagger} \right] \\ \text{Sub-level 3: } V_{x,\mu}^{(3)} &= \exp \left(\left[C_{x,\mu}^{(3)} U_{x,\mu}^\dagger \right]_{\text{TA}} \right) U_{x,\mu}, & C_{x,\mu}^{(3)} &= \frac{\epsilon_3}{6} \sum_{\pm\sigma \neq \mu} \left[V_{x,\sigma;\mu}^{(2)} V_{x+\sigma,\mu;\sigma}^{(2)} V_{x+\mu,\sigma;\mu}^{(2)\dagger} \right], \end{aligned} \quad (2)$$

where $[\dots]_{\text{TA}}$ is the traceless anti-hermitean part of its argument. Also, none of the directional indices are equal to each other. $C^{(n)}$ are staples constructed out of the smeared links in the $(n-1)$ -th sub-level weighted by the tunable smearing parameters, ϵ_n . Dropping the directional indices for the sake of brevity, as will be done quite often in Section IV, eq. (2) can be written in short as

$$V_x^{(n)} = \exp(A_x^{(n)}) U_x \quad \text{with} \quad A_x^{(n)} = \left[C_x^{(n)} U_x^\dagger \right]_{\text{TA}}. \quad (3)$$

The above construction is devoid of non-analytic operations, like projection to $\text{SU}(N_c)$. This means that $V^{(3)}$ can be expanded as a power series in U , however complicated the resulting expression might be. The one level HEX improved Dirac operator is given by replacing U with $V^{(3)}$ in eq. (1).

III. MANIFOLD, TANGENT VECTOR SPACE AND DERIVATIVES

In this section, we state some of the required results in Lie groups in the context of $\text{SU}(N_c)$. We refer the reader to [10, 11] for an extensive introduction to this topic. The group $\text{SU}(N_c)$ forms an $N = N_c^2 - 1$ dimensional manifold, with each point, U , on it being a group element. This means that there is a mapping, ϕ , from the neighbourhood of U to \mathbb{R}^N , called a coordinate chart. Let $f(U)$ be a real valued function defined over the manifold. We use the same notation f to refer to both f defined on the manifold as well as $f \circ \phi^{-1}$ defined on the coordinate chart.

We are interested in the fundamental representation of $\text{SU}(N_c)$, in which case, points on the manifold are $N_c \times N_c$ special unitary matrices. For any simply connected Lie group, the neighbourhood of any point is isomorphic to the neighbourhood of identity. The neighbourhoods are related by right-translating the elements in the neighbourhood of identity by U i.e.,

$$U' = \exp \left(i \sum_{A=1}^N \omega_A T_A \right) U, \quad (4)$$

where ω_A are real scalars. The matrix U' lies in the neighbourhood of U and $\exp \left(i \sum_{A=1}^N \omega_A T_A \right)$ lies in the neighbourhood of identity for small values of ω_A . The T_A are $N_c \times N_c$ traceless hermitian matrices satisfying $\text{Tr} T_A T_B = \delta_{A,B}$ and they are called the generators of $\text{SU}(N_c)$. Throughout this paper, the letter T (with or without super- and sub-scripts) denotes a traceless hermitian matrix. The group can be classified into families of 1-parameter abelian subgroups characterized by different T , each containing the elements $\exp(irT)$, for real r . This offers another way of finding the neighbourhood of U :

$$U' = \exp(irT) U. \quad (5)$$

The N -tuple, $(\omega_1, \omega_2, \dots, \omega_N)$, serves as a coordinate chart for the neighbourhood of U , enabling us to find derivatives of f (or rather of $f \circ \phi^{-1}$). Tangent vector space at U , denoted by $\mathbf{T}(U)$, is the vector space of directional

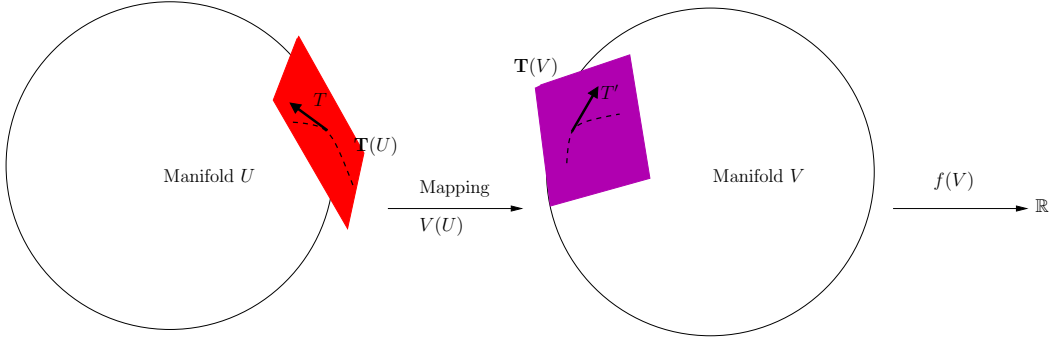


FIG. 1: Finding derivatives. The spheres on the left and the right schematically represent the $SU(N_c)$ manifolds whose points represent $SU(N_c)$ matrices U and V respectively. V is a function of U i.e., there is a mapping between the two manifolds. The red plane on the left represents the tangent vector space, $\mathbf{T}(U)$, while the magenta plane on the right is $\mathbf{T}(V)$. The matrix U changes along the dashed curve on the left and the tangent to the curve at U , in $\mathbf{T}(U)$, is along the direction T denoted by an arrow. This variation in U causes V to vary along the dashed curve on the right. The tangent to the curve at V is along the direction T' , in $\mathbf{T}(V)$. $f(V)$ is a real valued explicit function of V . As explained in the text, its derivative with respect to U is now an ordinary total derivative in the direction T' . What remains is to find T' .

derivatives tangent to the curves in \mathbb{R}^N passing through U . Usually, a tangent vector is defined as a directional derivative operator, $\sum_{A=1}^N v_A \partial / \partial \omega_A$, with v_A being real scalars. However, using the method of translation along a one parameter subgroup, the tangent vectors in $\mathbf{T}(U)$, denoted by $[\delta f / \delta U]_T$, become the ordinary total derivatives

$$\left[\frac{\delta f}{\delta U} \right]_T = \frac{d}{dr} f(e^{irT} U) \Big|_{r=0} \equiv \text{Tr} \left(T \frac{\delta f}{\delta U} \right), \quad (6)$$

where we have also implicitly defined the gradient, $\delta f / \delta U$. In this method, U varies along a curve, parametrized by r , on the manifold and we find the derivative along the tangent to this curve (one can think of this curve as being traced in the course of a molecular dynamics trajectory). This way of thinking is useful for the case of $V(U)$, which is an $SU(N_c)$ valued function of U . In this case, the curve traced by U maps to another curve traced by V on its manifold. Let the tangents to the two curves at U and V be along the directions determined by T and T' respectively. Let $f(V)$ be an explicit function of V . The derivative of $f(V)$ with respect to variation in U is a tangent vector at V along the direction T' :

$$\left[\frac{\delta f[V(U)]}{\delta U} \right]_T = \frac{d}{dr} f[V(e^{irT} U)] \Big|_{r=0} = \frac{d}{dr} f[e^{irT'} V] \Big|_{r=0}. \quad (7)$$

The final expression in the above equation is the tangent vector $[\delta f / \delta V]_{T'}$. This procedure is explained schematically in Figure 1. To implement the chain rule, now one has to find an ordinary derivative in $\mathbf{T}(V)$. The problem is to find the direction T' , which we will show to be very simple. This is to be contrasted with the usual method employed [7]:

$$\left[\frac{\delta f(V)}{\delta U} \right]_{T_A} = \frac{\partial}{\partial u_A} f[V(e^{iu_A T_A} U)] \Big|_{u=0} = \sum_B \frac{\partial v_B}{\partial u_A} \frac{\partial}{\partial v_B} f[e^{i \sum_C v_C T_C} V] \Big|_{u,v=0}. \quad (8)$$

This requires writing $\partial f / \partial v_B = \sum_{a,b} (\partial f / \partial V^{ab}) (\partial V^{ab} / \partial v_B)$, which involves derivatives with respect to each matrix element of V . In order to simplify further, we need derivatives of U and V with respect to r . These derivatives are defined as

$$\frac{dU}{dr} \equiv \frac{de^{irT} U}{dr} \Big|_{r=0} = iTU \quad \text{and} \quad \frac{dV(U)}{dr} = iT'V. \quad (9)$$

We shall write the above equation in short as $dU = iTU dr$ and call dU as the “variation in U ”. The derivative dU/dr is co-variant, as the assignment of a matrix U to each point on the manifold is unique up to a global transformation $G^\dagger U G$ with $G \in SU(N_c)$, thereby not requiring affine connections which are necessary when local transformations exist.

We end this section by noting a simple identity that will be very useful in Section IV: for any two matrices M and N , projection to the space spanned by the generators satisfies

$$\text{ReTr}(M [N]_{\text{TA}}) = \text{ReTr}([M]_{\text{TA}} N). \quad (10)$$

IV. CALCULATION OF FORCE

A trajectory of HMC [3] consists of an initial refreshment of momenta, $\Pi_{x,\mu}$, followed by classical evolution of gauge fields and their conjugate momenta through the simulation time, τ , according to a fictitious Hamiltonian, H . At the end of a trajectory, an acceptance step corrects for the discretisation error in the symplectic integrator. By definition, $\Pi_{x,\mu}$ determines the direction in which $U_{x,\mu}$ should evolve (refer eq. (9)). The Hamiltonian for staggered fermions is [12]

$$H = \frac{1}{2} \sum_{x,\mu} \text{Tr} \Pi_{x,\mu}^2 - S_g(U) - S_f \quad \text{and} \quad S_f = \frac{N_f}{8} \text{Tr} \ln (D^\dagger D), \quad (11)$$

where $S_g(U)$ is the gauge action and S_f is the fermion action. The momenta are evolved such that H is conserved. These conditions give the equations of motion

$$\frac{dU_{l,\mu}}{d\tau} = i\Pi_{l,\mu}U_{l,\mu} \quad \text{and} \quad \frac{d\Pi_{l,\mu}}{d\tau} = \frac{\delta S_f}{\delta U_{l,\mu}} + \frac{\delta S_g}{\delta U_{l,\mu}}. \quad (12)$$

$\delta S_f/\delta U$ and $\delta S_g/\delta U$ are called the fermion and gauge forces respectively. The fermion force in the case of the standard staggered action was derived in [12] and we give a slightly different derivation of it in Section IV A. In Section IV B, we derive the fermion force for HEX improved staggered action.

A. Fermion force for standard staggered action

The Dirac operator used in the standard staggered fermion action is a function of thin-links, $U_{x,\mu}$ (refer eq. (1)). When the link at site l varies in its tangent vector space by $dU_{l,\mu} = idrTU_{l,\mu}$, the fermion force is given by

$$\frac{8}{N_f} \left[\frac{\delta S_f}{\delta U_{l,\mu}} \right]_T = \text{Tr} \left(D^{-1} \frac{dD}{dr} \Big|_T + \text{h.c.} \right). \quad (13)$$

The derivative with respect to $U_{l,\mu}$ has been converted to an ordinary derivative with respect to r in its tangent vector space using eq. (9). By using the definition of D given in eq. (1), the ordinary derivative becomes

$$\frac{dD_{x,y}}{dr} \Big|_T = \frac{i}{2} \left(TU_{x,\mu} \delta_{x,y-\mu} \delta_{x,l} + U_{x-\mu,\mu}^\dagger T \delta_{x,y+\mu} \delta_{x-\mu,l} \right), \quad (14)$$

where we have used eq. (9). Having found the gauge derivative of D in one simple calculation, eq. (13) simplifies to

$$\frac{16}{N_f} \left[\frac{\delta S_f}{\delta U_{l,\mu}} \right]_T = \text{Tr} \left[iT \left(U_{l,\mu} D_{l+\mu,l}^{-1} + D_{l,l+\mu}^{-1} U_{l,\mu}^\dagger \right) + \text{h.c.} \right] \equiv \text{ReTr} [iT\Sigma_{l,\mu}], \quad (15)$$

where we have factored out T and collected the remaining terms as $\Sigma_{l,\mu}$. Taking the real part is superfluous. However we do so anticipating the simplifications in the next subsection. The fermion force follows by finding these derivatives along the generators. D^{-1} is usually evaluated by inserting a stochastic estimator of identity [12], but that does not concern our calculation.

B. Fermion force for HEX smeared staggered action

To avoid unnecessary complications, we restrict ourselves to one level of HEX improvement. The fermion action, S_f , is now an explicit function of the HEX smeared links, $V^{(3)}$ (refer eq. (3)). Having demonstrated the method of finding the ordinary derivative in tangent vector space for the case of the standard staggered action, we now demonstrate how it greatly simplifies the implementation of the chain rule required for hypercubic smearing schemes.

1. Essential simplification

The HEX smeared links in the n -th sub-level, $V_x^{(n)}$, are explicit functions of both thin-links and the smeared links in the $(n-1)$ -th sub-level. Any variation, $dU_{l,\mu} = idrT_A U_{l,\mu}$, in the thin-link at site l causes variations

$dV_x^{(n)} = idrT_x^{(n)}V_x^{(n)}$ in the smeared links (that are within the hypercube containing $U_{l,\mu}$). It is to be noted that we have dropped directional indices for the sake of brevity and ease of generalization to various sub-levels. Applying the product rule to eq. (3) and rewriting it in the form $idrT_x^{(n)}V_x^{(n)}$, leads to the expression

$$T_x^{(n)} = V_x^{(n)}U_x^\dagger T_A U_x V_x^{(n)\dagger} \delta_{x,l} - i \frac{d \exp(A_x^{(n)})}{dr} U_x V_x^{(n)\dagger}, \quad (16)$$

where we have replaced $\exp(A)$ by VU^\dagger to get to the above expression. Thus, we have determined the directions in $\mathbf{T}(V^{(n)})$, along which ordinary derivatives are to be found.

2. Recurrence Relation

The rest of the calculation proceeds backwards from the third sub-level by merely finding the ordinary derivatives with respect to r . Using the chain-rule (refer eq. (7)),

$$\left[\frac{\delta S_f}{\delta U_{l,\mu}} \right]_{T_A} = \sum_{x,\nu} \left[\frac{\delta S_f}{\delta V_{x,\nu}^{(3)}} \right]_{T_{x,\nu}^{(3)}}. \quad (17)$$

Since the HEX improved Dirac operator is obtained from the standard staggered Dirac operator by replacing U with $V^{(3)}$, we can simplify $\delta S_f / \delta V^{(3)}$ by borrowing results from Section IV A. After such replacements in eq. (15),

$$\frac{16}{N_f} \left[\frac{\delta S_f}{\delta U_{l,\mu}} \right]_{T_A} = \sum_{x,\nu} \text{ReTr} \left[iT_{x,\nu}^{(3)} \Sigma_{x,\nu}^{(3)} \right], \quad (18)$$

where $\Sigma_{x,\nu}^{(3)}$ is the HEX version of Σ . By using eq. (16) for $T^{(3)}$ and defining $\bar{\Sigma}^{(n)} \equiv V^{(n)\dagger} \Sigma^{(n)}$,

$$\frac{16}{N_f} \left[\frac{\delta S_f}{\delta U_{l,\mu}} \right]_{T_A} = \text{ReTr} \left[iT_A U_{l,\mu} \bar{\Sigma}_{l,\mu}^{(3)} V_{l,\mu}^{(3)} U_{l,\mu}^\dagger \right] + \sum_{x,\nu} \text{ReTr} \left[U_{x,\nu} \bar{\Sigma}_{x,\nu}^{(3)} \frac{d \exp(A_{x,\nu}^{(3)})}{dr} \right]. \quad (19)$$

The next step is to reduce the derivative of $\exp(A^{(3)})$ to a derivative of its exponent. Making a power series expansion of $\exp(A^{(3)})$, one would expect that a matrix $\Lambda^{(3)}$ can be defined such that,

$$\text{ReTr} \left[\Lambda_{x,\nu}^{(3)} \frac{dA_{x,\nu}^{(3)}}{dr} \right] \equiv \text{ReTr} \left[U_{x,\nu} \bar{\Sigma}_{x,\nu}^{(3)} \frac{d \exp(A_{x,\nu}^{(3)})}{dr} \right]. \quad (20)$$

For the case of $\text{SU}(3)$, the Cayley-Hamilton theorem leads to such a simplification by the expansion of $d[\exp(A)]$ in terms of A, A^2 and dA . This was done in [4] and using their result,

$$\Lambda_x = \text{Tr}(U_x \bar{\Sigma}_x B_1) A_x + \text{Tr}(U_x \bar{\Sigma}_x B_2) A_x^2 + f_1 U_x \bar{\Sigma}_x + f_2 A_x U_x \bar{\Sigma}_x + f_2 U_x \bar{\Sigma}_x A_x, \quad (21)$$

where $B_i = \sum_{j=0}^2 b_{ij} A^j$. The coefficients f_i and b_{ij} are complex scalar functions of the eigenvalues of A [4]. Using eq. (10) and the definition of A in eq. (3), it is possible to write $\text{ReTr}(\Lambda dA) = \text{ReTr}(\Lambda d[CU^\dagger]_{\text{TA}})$ as

$$\text{ReTr} \left[\Lambda_x^{(n)} \frac{d}{dr} \left[C_x^{(n)} U_x^\dagger \right]_{\text{TA}} \right] = \text{ReTr} \left[\left[\Lambda_x^{(n)} \right]_{\text{TA}} \frac{dC_x^{(n)}}{dr} U_x^\dagger \right] - \text{ReTr} \left[iT_A \left[\Lambda_x^{(n)} \right]_{\text{TA}} C_x^{(n)} U_x^\dagger \right] \delta_{x,l}. \quad (22)$$

Using eq. (2), we can schematically write $dC = d(V)VV + Vd(V)V + VVd(V)$ with the smeared links, V , which are one sub-level below. Each dV gives rise to a direction vector T , which can be cyclically permuted as the first term due to presence of the trace. Since the spatial and directional indices are summed over in eq. (19), these are dummy indices and we can factor out a T for each site x and direction. Reserving this calculation for Appendix A, we define a matrix Σ as

$$\sum_x \text{ReTr} \left[iT_x^{(n-1)} \Sigma_x^{(n-1)} \right] \equiv \sum_x \text{ReTr} \left[Z_x^{(n)} \frac{dC_x^{(n)}}{dr} \right] \quad \text{where} \quad Z_x^{(n)} = U_x^\dagger \left[\Lambda_x^{(n)} \right]_{\text{TA}}. \quad (23)$$

The matrix Σ arises naturally in this method leading to a very simple calculation (given in Appendix A) unlike the conventional method in [7] where derivatives of staple with respect to matrix elements of V are required. Using eq. (19), (22) and (23), after putting back the indices for the third sub-level,

$$\frac{16}{N_f} \left[\frac{\delta S_f}{\delta U_{l,\mu}} \right]_{TA} = \text{ReTr} \left[i T_A U_{l,\mu} \left\{ \bar{\Sigma}_{l,\mu}^{(3)} V_{l,\mu}^{(3)} - Z_{l,\mu}^{(3)} C_{l,\mu}^{(3)} \right\} U_{l,\mu}^\dagger \right] + \sum_{x,\mu,\nu} ' \text{ReTr} \left[i T_{x,\mu;\nu}^{(2)} \Sigma_{x,\mu;\nu}^{(2)} \right]. \quad (24)$$

\sum' indicates that none of the directions are equal. In the above equation, the second term in the right hand side has the same form as eq. (18), thereby giving us a recurrence relation. Along with the conditions that $V^{(0)} = U$ and $C^{(0)} = 0$, the rest of the terms can be written down. The equation for the fermion force becomes

$$\begin{aligned} \frac{16}{i N_f} \frac{\delta S_f}{\delta U_{l,\mu}} = & \left[U_{l,\mu} \left\{ \bar{\Sigma}_{l,\mu}^{(3)} V_{l,\mu}^{(3)} - Z_{l,\mu}^{(3)} C_{l,\mu}^{(3)} \right\} U_{l,\mu}^\dagger \right]_{TA} + \\ & \sum_{\nu} ' \left[U_{l,\mu} \left\{ \bar{\Sigma}_{l,\mu;\nu}^{(2)} V_{l,\mu;\nu}^{(2)} - Z_{l,\mu;\nu}^{(2)} C_{l,\mu;\nu}^{(2)} \right\} U_{l,\mu}^\dagger \right]_{TA} + \\ & \sum_{\nu,\rho} ' \left[U_{l,\mu} \left\{ \bar{\Sigma}_{l,\mu;\nu\rho}^{(1)} V_{l,\mu;\nu\rho}^{(1)} - Z_{l,\mu;\nu\rho}^{(1)} C_{l,\mu;\nu\rho}^{(1)} \right\} U_{l,\mu}^\dagger \right]_{TA} + \\ & \sum_{\nu,\rho,\eta} ' \left[U_{l,\mu} \bar{\Sigma}_{l,\mu;\nu\rho\eta}^{(0)} \right]_{TA}. \end{aligned} \quad (25)$$

If the action is improved by multiple levels of HEX, then $C^{(0)}$ is the staple constructed out of the smeared links one level below, which is $V^{(0)}$. Thus, the recurrence relation is easily extended to multiple levels.

V. MEMORY OPTIMIZED IMPLEMENTATION OF HYPERCUBIC SCHEMES

A code implementation of eq. (2) requires arrays U_μ and $V_{\mu;\nu}$ to store thin-links and the subsequent smeared links in various sub-levels respectively. Each of these arrays store 3×3 matrices at all lattice points and hence each of them is of size $9V_{lat}$. Since each step consists of updates done at all lattice points, we do not show the position indices of these arrays. In addition to them, we require arrays W_i for $1 \leq i \leq N_w$. These are required in order to provide work space to enable updating the same array $V_{\mu;\nu}$ with smeared links, when going from one sub-level to the next. Each of these arrays are again of size $9V_{lat}$. The problem is to minimize N_w . The first sub-level is easy to implement as it requires only thin-links, and the $V_{\mu;\nu}$ are empty to begin with. This step does not require any working arrays.

The smeared links, $V_{\sigma;\mu\nu}^{(1)}$, used in the second sub-level can also be written as $V_{\sigma;\eta}^{(1)}$, where η is the direction orthogonal to σ , μ and ν . With this observation, it is clear that both $V_{\sigma;\mu\nu}^{(1)}$ and $V_{\sigma;\eta}^{(2)}$ can be stored in the same array of the form $V_{\sigma;\eta}$. The brute force implementation of this sub-level would require $N_w = 12$, as two copies of V are required: one to store $V^{(1)}$ and the other for $V^{(2)}$.

A graphical representation of an algorithm to compute the second sub-level, so that N_w is reduced to 7, is given in Figure 2. The vertices (μ, ν) stand for the arrays $V_{\mu;\nu}$. Two vertices (μ, ν) and (ρ, η) are adjacent, if eq. (2) for $V_{\mu;\nu}^{(2)}$ involves $V_{\rho;\eta}^{(1)}$. If this is true, then by observation, $V_{\rho;\eta}^{(2)}$ also involves $V_{\mu;\nu}^{(1)}$ and hence this graph is not directed. At the beginning of the second sub-level, (μ, ν) contains $V_{\mu;\nu}^{(1)}$ at all lattice points. (μ, ν) cannot be updated with $V_{\mu;\nu}^{(2)}$ until the $V_{\rho;\eta}^{(2)}$ corresponding to all the adjacent vertices have been computed. The way to proceed becomes clear through the circular embedding of the graph. The algorithm is detailed in the caption of Figure 2. Here, we give a walk-through of the first two steps as follows. In the first step, we arbitrarily pick a vertex — we choose $(1, 2)$ in this example. The calculation of $V_{3;4}^{(2)}, V_{1;4}^{(2)}, V_{4;3}^{(2)}$ and $V_{1;3}^{(2)}$ requires $V_{1;2}^{(1)}$. Therefore, we first find $V^{(2)}$ for these adjacent vertices and store them in the working arrays W_1, W_2, W_3 and W_4 . Now, we are free to update $(1, 2)$ with $V_{1;2}^{(2)}$. This requires $V_{3;4}^{(1)}, V_{1;4}^{(1)}, V_{4;3}^{(1)}$ and $V_{1;3}^{(1)}$, which still remain untouched in the adjacent vertices. This is the reason for the specific order of these updates described in the figure. Once $(1, 2)$ is updated, no other vertex requires it and it gets disconnected from the graph. The second step proceeds similarly with respect to the vertex $(3, 4)$. However, at the end of the second step, $(3, 4)$ is updated to $V_{3;4}^{(2)}$, which was stored in W_1 during the first step. The rest of the steps of this algorithm proceed by repeating this procedure for a specific sequence of vertices, as shown in Figure 2, such that only 7 working arrays are required at any point of the algorithm. The total memory cost for the working space in terms of array size is $63V_{lat}$ compared to $108V_{lat}$ in the brute-force method. This cost can be further reduced by appealing to unitarity: only two rows of W_i at each lattice point are required. This reduces the memory cost further to $42V_{lat}$.

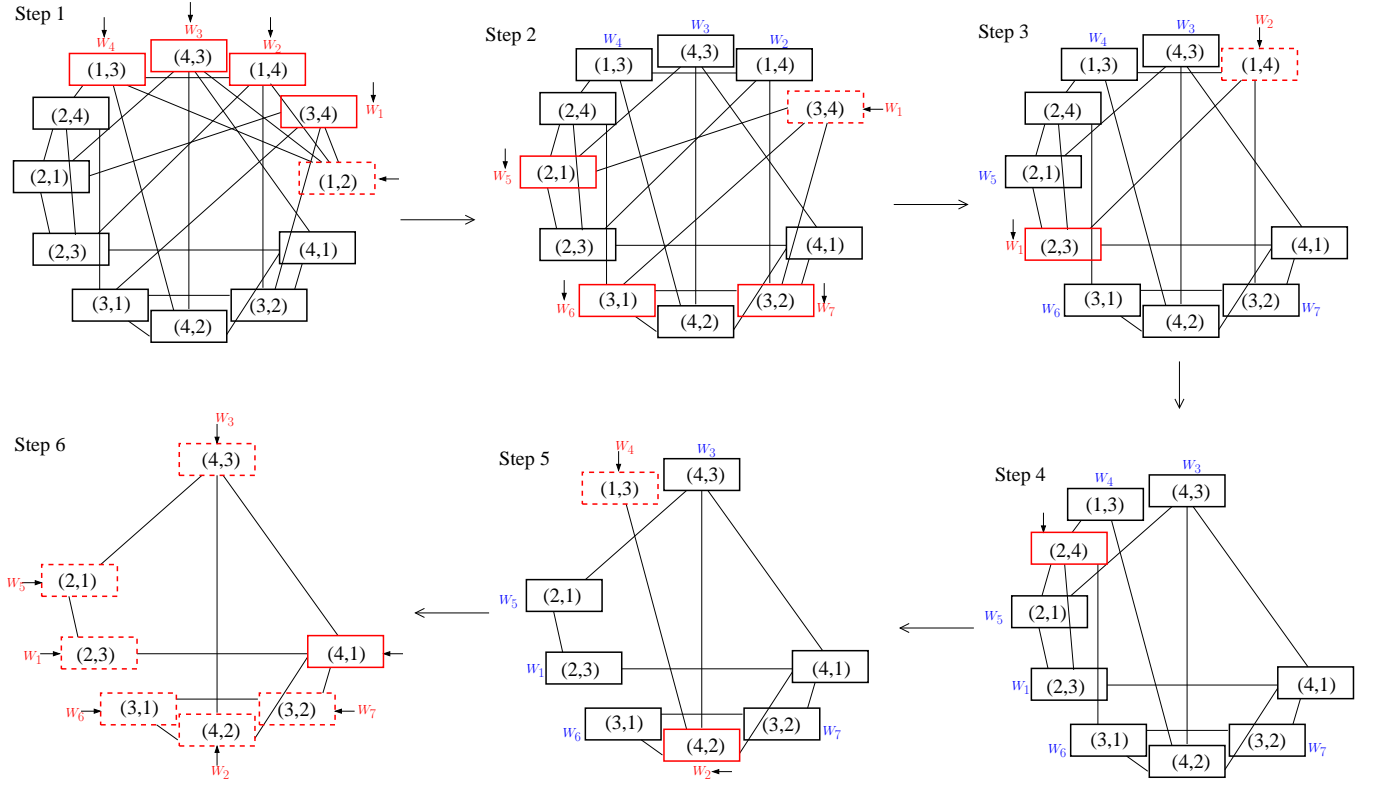


FIG. 2: Graphical illustration of memory optimized algorithm for the second sub-level of hypercubic schemes. The vertices (μ, ν) are the $9V_{lat}$ sized arrays, $V_{\mu;\nu}$, which stores 3×3 matrices at each lattice point. Two vertices, (μ, ν) and (ρ, η) , are connected by an edge (i.e., adjacent) if the construction of $V_{\mu;\nu}^{(2)}$ depends on $V_{\rho;\eta}^{(1)}$ (see eq. (2)), in which case $V_{\rho;\eta}^{(2)}$ also depends on $V_{\mu;\nu}^{(1)}$. There are 7 working arrays, W_i , which are indicated near some vertices and each of them are of size $9V_{lat}$. The algorithm proceeds from step 1 to 6 as shown in the figure. At the beginning of step 1, (μ, ν) contains $V_{\mu;\nu}^{(1)}$ at all lattice points. In each step, the red colored arrays are the ones which are relevant and the blue ones are dormant. Each step consists of the following operations. A short arrow with nothing at its tail, $\rightarrow W_i$ or $\rightarrow (\mu, \nu)$, indicates that $V_{\mu;\nu}^{(2)}$ is computed using $V_{\rho;\eta}^{(1)}$ contained in the vertices adjacent to (μ, ν) and stored in the array it points to. $W_i \rightarrow (\mu, \nu)$ means that data is copied from W_i to $V_{\mu;\nu}$. Solid or dashed red vertices indicate the order of these operations: the operations at the solid red vertices are done before the dashed ones, and amongst the solid vertices there is no hierarchy. Once $V_{\mu;\nu}$ is updated with $V_{\mu;\nu}^{(2)}$, the vertex and the edges attached to it are removed from the successive steps.

The third sub-level is again straight-forward. Only 4 working arrays are needed to compute $V_{\mu}^{(3)}$ (due to the 4 values of μ).

VI. SUMMARY

In this paper, we presented a simple method for implementing the chain rule to calculate fermion forces for the Hybrid Monte Carlo algorithm with HEX improved staggered action. It is based on the fact that the derivative of the action with respect to a thin-link can be written as the ordinary derivatives with respect to a real parameter in each of the tangent vector spaces of the smeared links. This way, we were able to avoid finding the derivatives with respect to all matrix elements of thin-links, as done in the literature [6, 7]. This has the obvious advantage of making the calculation very simple (Section IV B). In Section V, we gave an implementation of hypercubic schemes with less than half the memory requirement for work space when compared to the brute force implementation.

I would like to thank Prof. Sourendu Gupta for the discussions and for the idea of using the methods of differential geometry. I also thank him and Rahul Dandekar for careful reading of the manuscript.

Appendix A: Calculation of Σ

The definition of $\Sigma_{x,\mu}^{(3)}$ is given in eq. (18) and it involves the thin-link force with the replacement of thin-links with the smeared links, $V^{(3)}$. For $n = 0, 1$ and 2 , $\Sigma^{(n)}$ is defined by the relation

$$\sum_x \text{ReTr} \left[iT_x^{(n)} \Sigma_x^{(n)} \right] \equiv \sum_x \text{ReTr} \left[Z_x^{(n+1)} \frac{dC_x^{(n+1)}}{dr} \right]. \quad (\text{A1})$$

From eq. (2), $C^{(n+1)}$ is constructed out of products of three $V^{(n)}$. Using the product rule and then the cyclicity of the trace, we take all dV to the front and dV^\dagger to the end. Due to summation over all spatial indices, we translate each term such that dV or dV^\dagger in all the terms are at position x . Due to summation over all the directional indices, these are dummy indices and can be interchanged. dV and dV^\dagger bring a factor of $T_x^{(n)}$ to the front by this construction. We give the result for the second sub-level:

$$\begin{aligned} \frac{6}{\epsilon_3} \Sigma_{x,\nu;\sigma}^{(2)} = & V_{x,\sigma;\nu}^{(2)} V_{x+\sigma,\nu;\sigma}^{(2)} V_{x+\nu,\sigma;\nu}^{(2)\dagger} Z_{x,\nu}^{(3)} + V_{x,\sigma;\nu}^{(2)} V_{x-\nu+\sigma,\nu;\sigma}^{(2)\dagger} Z_{x-\nu,\sigma}^{(3)} V_{x-\nu,\nu;\sigma}^{(2)} \\ & - Z_{x-\nu,\nu}^{(3)} V_{x-\nu,\sigma;\nu}^{(2)} V_{x-\nu+\sigma,\nu;\sigma}^{(2)} V_{x,\sigma;\nu}^{(2)\dagger} - V_{x,\nu;\sigma}^{(2)} V_{x+\nu,\sigma;\nu}^{(2)} Z_{x+\sigma,\nu}^{(3)} V_{x,\sigma;\nu}^{(2)\dagger} \\ & + V_{x,\sigma;\nu}^{(2)} V_{x+\sigma,\nu;\sigma}^{(2)} Z_{x+\nu,\sigma}^{(3)} V_{x,\nu;\sigma}^{(2)\dagger} + V_{x,\sigma;\nu}^{(2)} Z_{x+\sigma-\nu,\nu}^{(3)} V_{x-\nu,\sigma;\nu}^{(2)\dagger} V_{x-\nu,\nu;\sigma}^{(2)}. \end{aligned} \quad (\text{A2})$$

It is straight forward to use appropriate directional indices to get $\Sigma_{x,\nu;\sigma\rho}^{(1)}$ and $\Sigma_{x,\nu;\sigma\rho\eta}^{(0)}$. Using these, we can calculate $\overline{\Sigma} = V^\dagger \Sigma$.

-
- [1] J. Polonyi and H. W. Wyld, Phys. Rev. Lett. 51, 2257 (1983).
 - [2] G. G. Batrouni *et al.*, Phys. Rev. D 32, 2736 (1985).
 - [3] S. Duane, A. D. Kennedy, B. J. Pendleton and D. Roweth, Phys. Lett. B 195, 216 (1987).
 - [4] C. Morningstar and M. J. Peardon, Phys. Rev. D 69, 054501 (2004).
 - [5] S. Capitani, S. Dürr and C. Hoelbling, *J. H. E. P.* 0611, 028 (2006).
 - [6] W. Kamleh, D. B. Leinweber and A. G. Williams, Phys. Rev. D 70, 014502 (2004).
 - [7] S. Durr *et al.*, *J. H. E. P.* 08, 148 (2011).
 - [8] A. Hasenfratz, R. Hoffman and S. Schaefer, *J. H. E. P.* 05, 029 (2007)
 - [9] A. Hasenfratz and F. Knechtli, Phys. Rev. D 64, 034504 (2001).
 - [10] M. Daniel and C. M. Viallet, Rev. Mod. Phys. 52, 175 (1980).
 - [11] R. Gilmore, *Lie Groups, Lie Algebras and Some of Their Applications*, John Wiley & Sons, New York (1974).
 - [12] S. Gottlieb *et al.*, Phys. Rev. D 35, 2531 (1987).